Multimedia Tools and Applications, 9, 275–302 (1992)

Design and Configuration Rationales for Digital Video Storage and Delivery Systems*

CYRIL U. ORJI orji@fiu.edu
School of Computer Science, Florida International University, Miami, FL 33199, USA.

DONALD A. ADJEROH donald@cs.cuhk.edu.hk
Department of Computer Science and Engineering, The Chinese University of Hong Kong, HONG KONG.

PATRICK O. BOBBIE bobbie@cis.famu.edu
Department of Computer and Information Systems, Florida A&M University, Tallahassee, FL 32307, USA.

KINGSLEY C. NWOSU nwosuck@harpo.wh.lucent.com
Lucent Technologies, (AT&T Bell Labs), 67 Whippany Road, Whippany, NJ 07981-0903, USA.

Abstract. Recent advances in computing technology have brought multimedia information processing to prominence. The ability to digitize, store, retrieve, process, and transport analog information in digital form has changed the dimensions of information handling. Several architectural and network configurations have been proposed for efficient and reliable digital video delivery systems. However, these proposals succeed only in addressing subsets of the whole problem. In this paper, we discuss the characteristics of video services. These include Cable Television, Pay-Per-View, and Video Repository Centers. We also discuss requirements for "Video On Demand" services. With respect to these video services, we analyze two important video properties: image quality and response time. We discuss and present configurations of a Digital Video Delivery System (DVDS) from three general system components - servers, clients, and connectivities. Pertinent issues in developing each component are also analyzed. We also present an architecture of a DVDS that can support the various functionalities that exist in the various video services. Lastly, we discuss data allocation strategies which impact performance of interactive video on demand (IVOD). We present preliminary results from a study using a limited form of mirroring to support high performance IVOD.

Keywords: video services, video delivery, video storage, video-on-demand, video servers, delivery systems, data allocation

1. Introduction

The ability to digitize, store, retrieve, process, and transport analog information has changed the dimensions of information handling in the past few years. This is a consequence of advancement in computing technology. A by-product of these advances is the emergence of multimedia information processing. Multimedia encompasses the integrated generation, representation, processing, storage, and dissemination

* This work has been supported in part by grants from NASA (NAGW-4080), ARO (BMDO grant DAAH04-0024), ARO (ADSRC DAAH04-95-10250) and NSF (IRI-94099661).

of independent machine processable information. Multimedia information processing becomes more useful when Multimedia Data Base Systems (MDBSs) become functionally feasible. MDBSs require full support of the operations of conventional Data Base Management Systems (DBMSs), with respect to query operations that involve joins, selects, projections, etc. As the research and development in MDBSs continue, the available and applicable technologies are being exploited to handle other multimedia-related activities. Prominent among them are the digitization and transport of video data. These have been made possible largely as a result of advances in video coding technologies [1, 11, 12, 14, 16, 2], VLSI, and network technologies which have provided fast CPUs, fast and large bandwidth networks, and interconnectivities. Furthermore, efficient compression techniques have helped to reduce the size of digitized objects to manageable sizes without adversely affecting the quality of video objects.

The ability to digitize and store video information has changed the way business and entertainment industries provide information to their clients/customers. With intelligent and cost effective configurations, a cable company, for example, can store a large number of movies and programs for its customers on-demand basis. The need to perform conventional VCR functions like forward, backward, pause, etc., on digital video data have opened up research efforts on interactive video on demand (IVOD). Various architectural alternatives [9, 25] and data placement algorithms [4, 23] have been used to address this problem.

Given the availability of hardware technologies for building Digital Video Delivery Systems (DVDSs), research and development efforts are being invested in reliable and efficient processing techniques as well as cost-efficient architectural configurations [8, 19, 20, 25]. For example, Video On Demand (VOD) and near video on demand (NVOD) were described in [13] with the intent to identify as much common equipment as possible that will be compatible across the various applications. More recently, the authors in [7] adapted the basic architecture for interactive television (ITV) suggested in [24] and used it as a framework for the development of a hierarchical configuration of multimedia servers and network switches for ITV. Their system consists of (a) information providers such as entertainment houses and television stations; (b) network providers that transport media over integrated networks; and (c) several levels of storage providers that manage data storage in multimedia servers and that contain network switches.

The framework developed in [7] appears to have focused primarily on ITV, and on solving specific technological problems. In this paper, we take a more global look at the problem and develop a framework for an integrated environment for handling all the conventional video services.

The remainder of this paper is organized as follows. In Section 2 we discuss the characteristics of the prevailing video services available to the customers such as Cable Television, Pay-Per-View, and Video Repository Centers. In Section 3 we discuss the requirements of Video On Demand services not currently fully available to video customers. We analyze two important video properties - image quality and response time - with respect to video services. In Section 4 we discuss and present

Table 1. **Video services and major characteristics**

Type of Service	Characteristics		
	Program Availability	Fast Forward/ Reverse	Cost of Service
CATV	Only at designated times; Once a day	No	Flat rate, independent of programs viewed.
PPV	Only at designated times; More than once a day	No	Sliding rate; based on program viewed.
VRC	Always, if program is on tape	Yes	Rental cost
VOD	Any time	Yes	Not clearly defined. Possibly based on amount of data.

configurations of a DVDS from three general system components: server, clients, and connectivities. We analyze pertinent issues in developing each component and present an architecture of a DVDS whose goal is to provide the functionalities that exist for video customers and other expected capabilities. An important parameter in the design of a server is the data allocation scheme employed in the server(s). This has been shown to greatly affect performance of video-on-demand systems. We devote Section 5 to the discussion of two popular data allocation algorithms, *Distributed Cyclic Layout (DCL)* and *Staggered Distributed Cyclic Layout (SDCL)*. In Section 6 we present preliminary results of an ongoing study aimed at exploiting the characteristics of DCL and SDCL in a limited mirroring scheme to improve performance of an IVOD system. Our concluding remarks are presented in Section 7. The architectures described in this paper do not apply to Distributed Data Base Systems that support data base operations. They specifically address digital audio/video delivery services.

2. Video Services

In order to get a better understanding of the requirements of a Video Delivery System (VDS), we first discuss the various types of services provided by such systems. The classifications of the services are primarily based on the flexibility of control by which a user can obtain specific video entities or sessions, and the temporal implications. We account for the services that are currently available to users and the services that should be available with the advent of improved digital information processing technologies. Below we discuss the various video services, and the differences between them. A summary of their distinguishing characteristics is presented in Table 1.

2.1. Cable Television (CATV) service

Subscribers to Community Antenna Television (CATV) (or simply Cable Television), obtain their video information by channel selection through a TV at specified times. The services are broadcast to the subscribers from a video source and special devices may be required to block unauthorized access. The services are multicast to the users who must tune to the appropriate channel at designated times to view a specific program. Users can tune to a particular program during its transmission, but they will be unable to view the program after its transmission unless it is being re-broadcast at another time. The re-broadcast for CATV is infrequent or seldom, consequently, as a distinguishing characteristic, we will assume, without loss of generality, that CATV programs are broadcast once a day. The CATV service customers are usually charged flat rates, therefore, whether the customer viewed any programs or not, the customer must pay for the service.

2.2. Pay-Per-View (PPV) service

Subscribers to PPV services have more viewing opportunities than the CATV customers because programs are broadcast more than once a day. Similar to CATV service, PPV programs are broadcast at specific times. Due to the frequent rebroadcast, the PPV service gives its customers flexibility to view a particular program from its scheduled times. Unlike the CATV customers, the PPV customers are charged per program viewed. After a customer has been given access to view a program, whether the customer actually views the program or not, the customer must pay for the service.

2.3. Video Repository Center (VRC) service

Another service option available to customers is the VRC service. In this option, the customer obtains desired video tapes from a video rental center, at a per tape charge. A video rental center stores a number of recorded events (such as movies, sports, and music videos) and rents them to the public. Normally, there is no limit to the number of tapes that a customer may rent, however, as a business practice, a video rental center may impose a limit on the number of tapes that may be rented out by a customer. VRC service customers are expected to return the rented tapes after some specified time. Besides renting video tapes from VRCs, one can also purchase them. One interesting aspect of VRC service is that for certain programs, the producers may make the business decision to not make them available on tape. Furthermore, viewing recorded versions of some events may take away the element of realism, suspense, or simultaneity. This is more so when we consider the case of movies where it is generally a common business practice to delay making new movies available on tape until such movies have had reasonable tenure in movie theaters.

2.4. Video-On-Demand (VOD) service

In a VOD service, a customer has full control of what video program to see, when to see, and how to see it. It offers a customer all the services available in CATV, PPV, and VRC services with much more flexibility. Under the VOD service, a user has the capability to request for any available program, real time or not, at any time, and starting at any specified duration of the event (in the case of recorded events). In an ideal case, a VOD service is expected to provide functionalities available in a VCR service, like fast-forward, rewind, skip, pause, and so on. However, this is an active research area, and in Section 5 we describe a data-layout strategy that facilitates the provision of these services.

The assumption in VOD is that unlike the VRC service, a customer should not leave his/her domicile in order to obtain the required services. It, therefore, means that programs are not necessarily multicast to customers. In general one can assume that programs are narrowcast to customers based on their temporal requirements.

To a user, a video program is coming from a single source. The underlying distributed or parallel object storage is transparent. Users can view real-time events, if they choose, or they can view them at later times, in any way they choose. Since there is a control and management site that extracts the required video segments for the customers, the customers can be billed based on the amount of video data viewed (this is possible, but may not be feasible); and charges are not based on a flat rate or per program.

One of the major limiting constraints on VOD is the ability to satisfy the huge bandwidth and capacity requirements of VOD. Currently, arrays of inexpensive disks (RAID) [5] and high performance servers with large switching capacity to instantly connect data from disks to I/O channels are used to address these problems.

3. Video Delivery Properties

Problems associated with data collection, storage, transmission and delivery must be addressed in the process of video delivery. To address the huge data volume problem, different *compression* and *encoding* standards have been proposed and utilized[17, 18, 27]. For example, the first specification from the Moving Picture Experts Group, MPEG-1, uses interframe compression to achieve compression ratios of about 50:1 by applying lossy methods. The MPEG specification utilizes *differential coding* techniques based on *temporal reduction*, also known as *inter-frame prediction*. Under this technique, corresponding blocks of consecutive images are compared and only the differences are coded based on some parameters. Other specifications of MPEG achieve as high as 200:1 compression ratio. Consequently, instead of a noncompressed digitized video image requiring a data rate of about 27 Mbytes/s to transfer a screen of 640 by 480 pixels, with 24 bits per pixel, at 30 frames per second, MPEG1 reduces the requirement to about 550 kB/s. Prior to the advent of MPEG, the specification for photo-quality still images developed by the Joint Photographic Experts Group (JPEG) was used for full-motion video. It

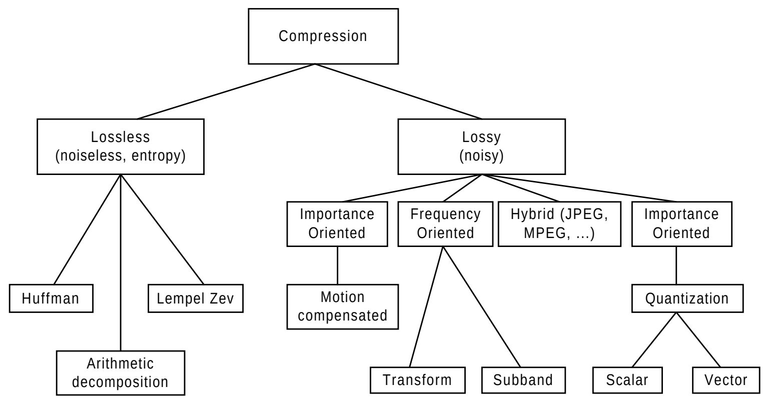

Figure 1. **Synopsis of prevailing coding techniques.**

runs at 30 frames per second and does not use differential coding technique. JPEG can achieve a lossless compression ratio of 50:1. Figure 1, taken from [6] and [26], gives a synopsis of the prevailing coding techniques.

Table 2. **Examples of video object throughput requirements**

Data Type	**Data Rate (per sec.)**
Digitized Audio	640 Kbits
PAL Digitized Video	30 Mbytes
MPEG-I Video	1.5 Mbits to 3.0 Mbits
MPEG-II Video	3.0 Mbits to 6.0 Mbits
NTSC Digitized Video	25.2 MBytes
HDTV	144 MBytes

Most video objects require a minimum amount of data utilization per unit time. The data utilization rate is the amount of data consumed per unit time to satisfy operating requirements. For example, if a television display requires x frames per second, and if each frame is y bits, then, the data utilization rate is $x.y$ bits/sec. It is, therefore, necessary that this amount of data be available when needed for the object to be meaningfully displayed. Table 2 shows examples of some of the throughput requirements for some types of video objects and standards such as compressed MPEG I & II, PAL (Phase Alternation by Line – European Television Standard), NTSC (National Television System Committee) and HDTV (High Definition TV).

Different types of networks may be required to connect customers serviced by a VOD system. Figure 2 shows a possible configuration of a VOD system. Such a system includes Wide Area, Metropolitan Area, and Local Area Networks. Buffering is often used to mask the high latency associated with these networks. Buffers can be used both at the client and server ends.

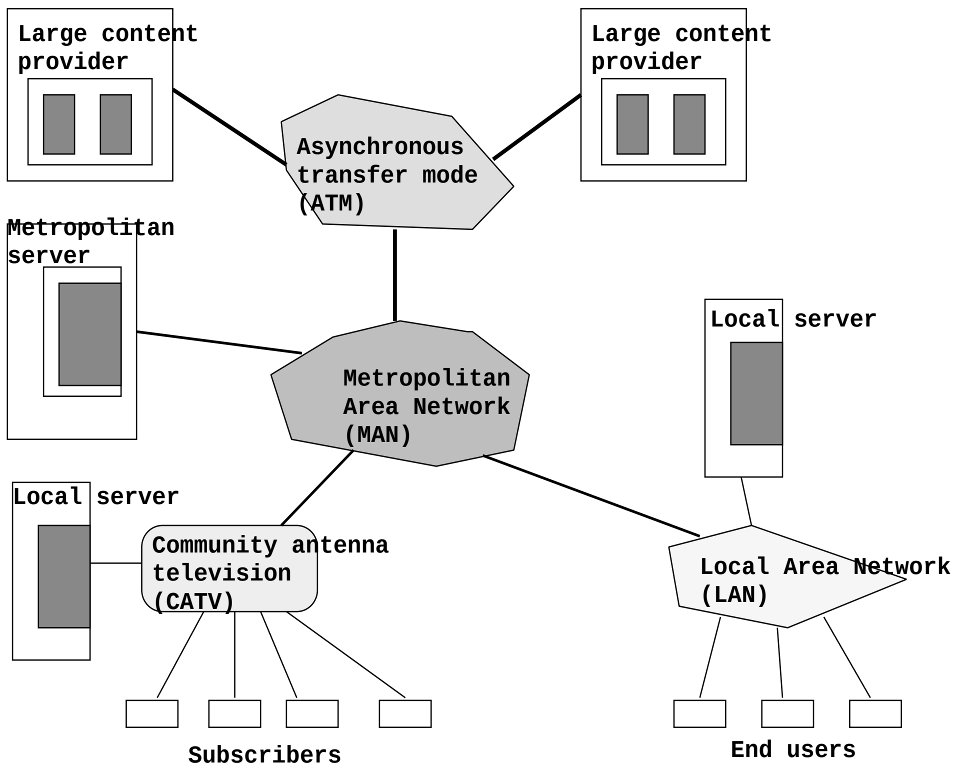

Figure 2. **Basic architecture of a digital video delivery system.**

The process of data collection, compression, storage and delivery is usually transparent to the end user. The VOD customer is usually concerned with the *quality of an image* and the *response time* for video delivery. We briefly describe these two important user properties.

3.1. Quality of image

Most multimedia applications can absorb the loss of a few frames per second without the viewer observing a noticeable degradation in the quality of the video. An important parameter that impacts video quality is the video coding rate. Increasing the coding bit rate for digital video improves the image quality, but also increases the amount of data that must be handled. The transport and synchronization of data at a display site are a few of the many problems of continuous media. However, recent advances in technology have been helpful in mitigating the problems.

3.2. Response time

Another important issue to video customers is the delay between requesting a video program and getting the requested video scenes. For all practical purposes, response time depends largely on the type of service requested. In the case of the CATV, the customer has no option on program availability since programs are usually scheduled for specific number of times (usually once per day) at inflexible times. A customer can only view a program at the scheduled time. Once a CATV program is being broadcast, the response time becomes equivalent to the delay necessary for changing from one channel to another.

For PPV, on the other hand, a given program is scheduled more frequently over a 24-hour period than a CATV program. In that case, given the static and pre-knowledge of the showing times of PPV programs, the response time is usually minimal relative to the scheduled time. A time delay, in the magnitude of a few minutes, is usually permissible.

Due to the complexity of VOD, its response time is greatly affected by a number of factors. However, the response time is expected to be in the range of the CATV and PPV response times. In other words, it is expected that the response time may be slightly worse than CATV response time, but should be better than PPV response time. Since different customers may request the same or different programs at any given time, the response time is greatly affected by the technology of the storage and delivery systems among other factors. As an example, consider a local multimedia server serving a neighborhood. For this example adapted from [7] (See Figure 3), due to disk I/O bandwidth limitation, only 120 out of the 1000 households can simultaneously receive movies-on-demand. The number of concurrent subscribers can be increased by using one of the following techniques:

- Increasing the number of disks which will increase the I/O throughput.
- Transmitting video that have a lower data rate demand. For example, instead of transmitting 16-Mbps HDTV movies, 3- to 6-Mbps MPEG-2 movies can be transmitted.
- Using techniques that increase the capacity of VOD systems such as segmentation and multicasting.

The response time for VRC requests is very different from its counterparts. For one, the customer has to make a trip to the video center, and secondly, he/she has to either wait in line or wait for tape availability. Thus, considerable amount of time may elapse from the time one wishes to view a program and the time that one actually starts viewing the program. Sometimes, a customer may have to visit more than one VRC in order to obtain a frequently requested tape. Therefore, an implicit assumption with obtaining videos from VRC is that a customer is willing to expend the possible considerable time-delay involved.

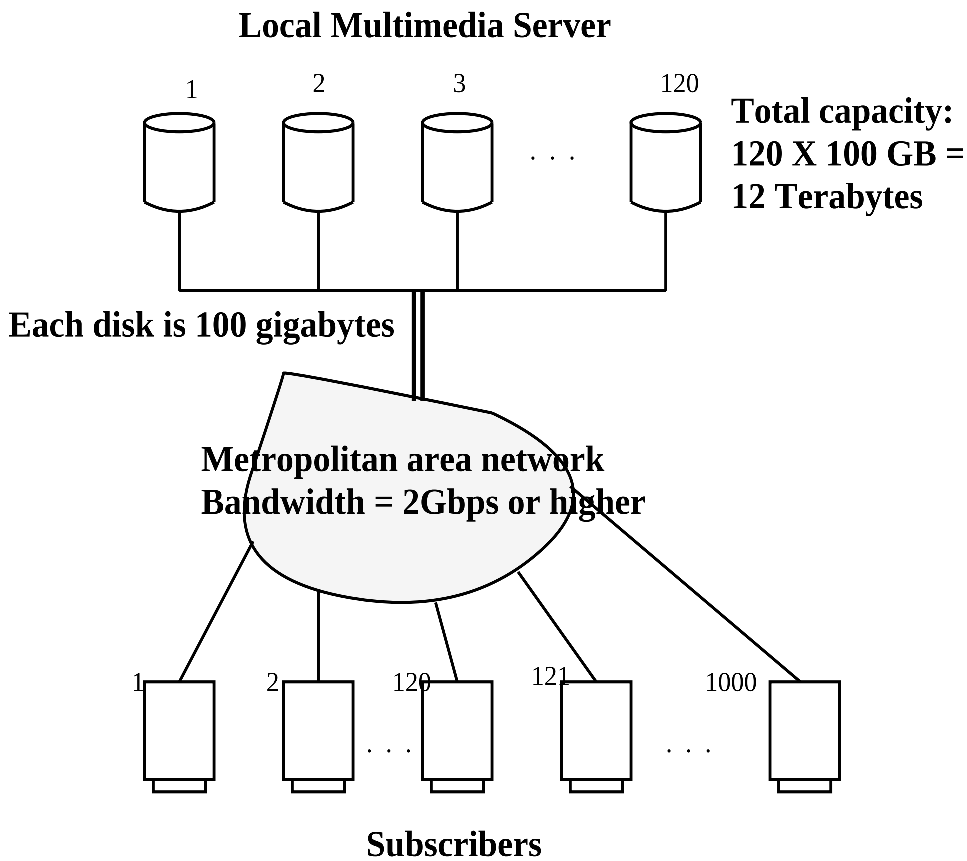

Figure 3. **A local multimedia server serving a neighborhood of 1,000 households. Server provides HDTV-type movies-on-demand; only 120 users can receive movies-on-demand simultaneously. (Adapted from [7].)**

4. Digital Video Delivery Systems

One of the major objectives of a DVDS is to satisfy customer requests more efficiently than the other delivery services. A DVDS must be capable of delivering pre-scheduled programs to subscribers (like the CATV); moreover, it must afford its customers the opportunity to view PPV-type programs. It is expected that a DVDS will provide users the flexibility of viewing programs that are not frequently requested at a time-delay that on the average, should be much less expensive than the time for a trip to a VRC.

The DVDS architecture

There are many possible hardware configurations for a DVDS system. However, in this paper, we are concerned mostly with technical implications of the subsystems and sub-components. Without loss of generality, we classify the DVDS components into servers, clients, and connectivities.

4.1. The DVDS server

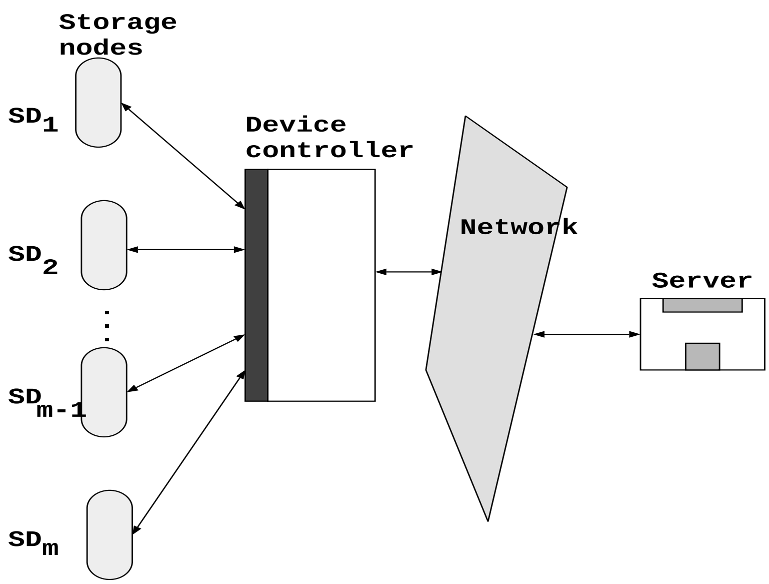

Figure 4. **A single machine video server**

A DVDS server is the subsystem that is responsible for the storage and retrieval of digital data. In addition, the server subsystem may be responsible for running application programs and utilities for managing customer accesses to stored data. Typically, the server will only be responsible for up-loading and down-loading data to machines that perform other functions. The server component may be a single machine or a collection of machines acting with a single system image. In either case, the server comprises a number of storage devices. For example, Figures 4, 5, and 6, show different configuration alternatives for a video server. Figure 4 shows the simplest server configuration which consists of a single machine with a number of storage devices attached to it. Usually, a DVDS should be able to accommodate hundreds or even thousands of video programs on its storage devices and also be able to handle concurrent video streams. However, a single machine is limited not only by its bandwidth but also by the number of peripheral devices, such as disk drives, that can be attached to it.

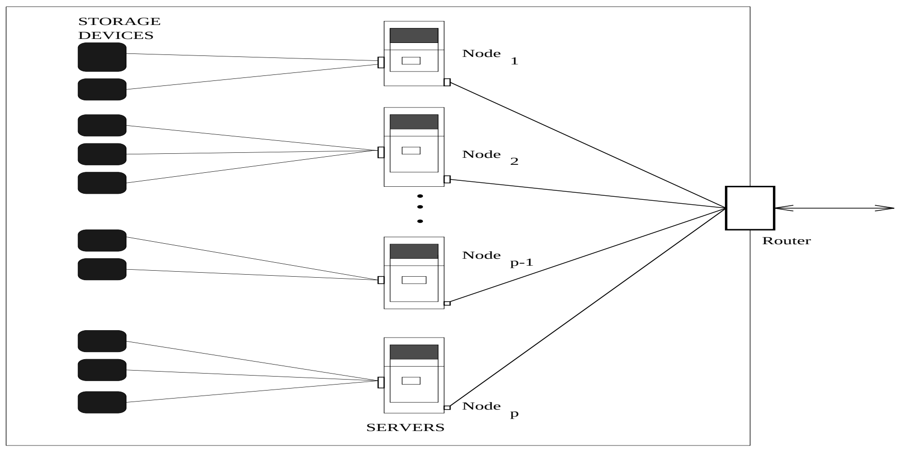

Figure 5. **Multiple video servers**

An alternative server configuration is shown in Figure 5. This comprises a cluster of machines, each with, possibly, some local storage devices. A request for data is directed to one of the servers. Provided the server is available, it is completely responsible for servicing the user request. A cluster of servers is usually configured such that any given request can be satisfied by any of the server nodes. To accomplish that, each node is able to access both remote and local storage devices where the video programs may be stored. A local area network (LAN) or an interconnection network between node partitions is used to accomplish this.

The former approach necessitates a global data management strategy. However, given the morphology of LANs, the traffic and communication overhead will greatly affect the temporal data availability requirements. The latter approach, as shown in Figure 6, partitions the available machines into two partitions, OUTBOUND and INBOUND nodes interconnected via an interconnection network. The OUTBOUND nodes ($ONode_1, \ldots, ONode_p$) are responsible for accepting data requests, determining their locations and sending the requests to one of the INBOUND nodes ($INode_1, \ldots, INode_q$). The INBOUND nodes are responsible for storing and retrieving data. The global data management subsystem is configured in the OUTBOUND nodes so that each of the nodes can access any of the stored data through the appropriate INBOUND nodes.

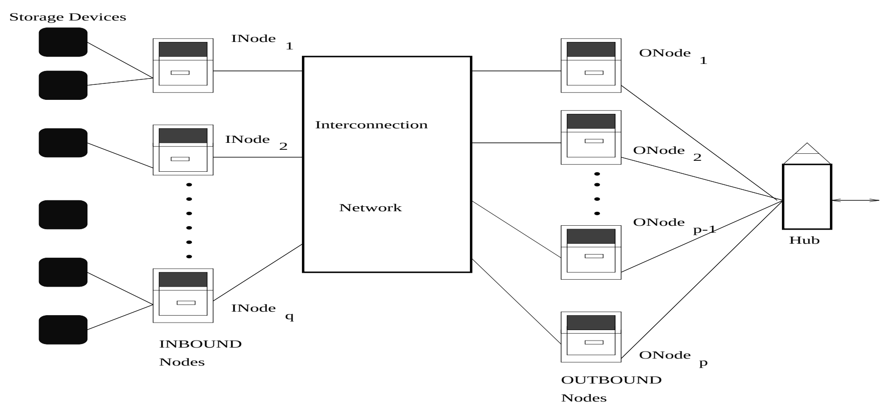

Figure 6. **Partitioned video servers**

4.1.1. Data allocation issues

One of the most important issues in data storage and retrieval is achieving the necessary data availability rates to support customer requests. A related problem is the distribution of concurrent customer request streams to different nodes. In [21], some detailed discussions and solutions were proposed for the relative speed disparity between the storage devices and other computing system components. Specifically, the speed of storage devices have not improved relative to the speeds of CPUs, networks, etc.; therefore, in instances like multimedia applications, the speed of the storage device (also network transmission delay) rather than the speed of the CPU is the limiting factor. We analyze the storage and retrieval problem from single-server and multiple-server viewpoints.

Assume a single-server system that supports n concurrent video streams. Let each stream require a data rate of r Mb/s. Therefore, the data retrieval bandwidth is rn Mb/s. In the presence of MPEG-I data encoding rate (about 1.5 Mb/s) and hundreds of concurrent streams, it becomes obvious that using one storage device to achieve the bandwidth is very difficult. Therefore, it is necessary to utilize multiple storage devices configured for parallel access. Under this configuration, a single video object can be decomposed and stored in multiple storage devices for parallel retrieval. This is usually accomplished by *intra-node striping*. Intra-node striping occurs when an object is decomposed into segments that can be stored individually on different storage devices attached to a single machine such that the whole object can be retrieved in one unit I/O.

One of the problems with intra-node striping is that the cumulative bandwidth of video streams may exceed the system bandwidth. This usually results in significant delays in satisfying I/O requests. A good way of handling the problem is to utilize multiple nodes with *inter-node striping*. An object is inter-node striped when it is decomposed into a number of segments and each segment is allocated to storage devices that are attached to mutually exclusive INBOUND nodes (see Figure 6).

Within each INode, each segment can undergo intra-node striping. Consequently, the task of retrieving an entire object is spread across multiple nodes. The number of storage devices configured for the INBOUND nodes is largely determined by the bandwidth of a given number of video streams. Consider an environment where 1000 MPEG-1 streams requiring about 200 $MBytes/s$ typify the workload; this throughput requirement becomes an important factor when the number of storage devices configured as INBOUND nodes is determined.

4.1.2. Analysis of configuration and data allocation schemes

In this section, we analyze different configuration and data allocation schemes with respect to the number of customers that can be concurrently supported given response time implications. At the time of this paper, the maximum capacity of commercially available disk drives is about 9 Gbytes. However, to simplify analysis, we assume that a 100 minute video (size of ≈12.6 Gbytes) is stored in one *logical* disk. To further simply presentation, we view this logical disk as a single disk with a transfer rate of about 1 Mbyte/sec. Furthermore, we assume that s_i, the ith segment of a video object (of n segments), is retrieved per unit time. The time that the retrieval of a video object is initiated is represented as t_0 while the time that the display starts (start-up time) at the customer site is represented as z_0. Also, we let $z_0 - t_0 = \epsilon$, and ignore network latency.

Single Server, Single Storage Device (SSSSD)

This is the simplest configuration for a VOD system. However, the retrieval throughput of the system is limited by the transfer rate of the storage device, i.e., 1 Mbyte/s. If the minimization of the response time is a primary objective, then at most one customer can be supported at any given time. By concurrent customer access we mean the viewing of the same or different video objects initiated at different times. More than one customer can be supported at considerable response time degradation. In any event, two seconds of data retrieval is needed for one second data consumption. Therefore, s_i is retrieved at t_i but consumed at t_j such that $t_j > t_i$. Specifically, at z_k $(k > 0)$, the segments consumed are s_{2k-1} and s_{2k}, which must have been retrieved at t_{2k-1}, t_{2k}, respectively. Therefore,

$$\forall_i \ (i = 1, \ldots, n) \ t_i < t_{i/2} + \epsilon \quad \equiv \quad \forall_i \ (i = 1, \ldots, n) \ i < \frac{i}{2} + \epsilon \tag{1}$$

A realistic solution lies on the retrieval and display of the last segment. Therefore, z_0 really depends on

$$n < \frac{n}{2} + \epsilon \tag{2}$$

For example, Figure 7 shows the retrieval and consumption times for 10 segments of a video object. Note, however, that in practice, this option may not be very attractive given the amount of buffers needed to implement it.

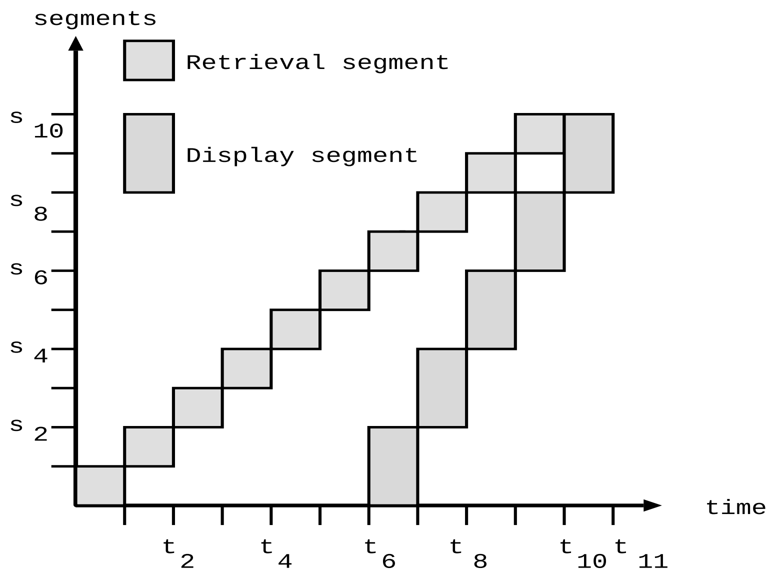

Figure 7. **Sample retrieval and display times of video segments**

Single Server, Multiple Storage Devices (SSMSD)

An SSMSD VOD server consists of one server machine with a number of storage devices attached to it. Assume that there are m storage devices. When a video object is wholly contained within a storage device, then the conditions established in SSSSD systems apply, thereby, limiting the number of customers that can be concurrently serviced to m. In each customer request, the response time, with respect to z_0, established in the SSSSD case applies.

On the other, the video objects can be striped in such a way that each object is stored within two storage devices. In this case, in a unit time, the two storage devices can concurrently retrieve two segments of the object for display in the next unit time. This reduces the start-up time to a unit time. Formally,

$$\forall_i \ (i = 1, \ldots, n) \ i + 1 \leq i + \epsilon \implies \epsilon \geq 1 \implies z_0 = 1 \tag{3}$$

Figure 8 shows the retrieval and consumption times for a two-way striped, 8-segment video objects (O_1, O_2). It shows that the segments of one object are displayed concurrently with the retrieval of the segments of the second object. This technique is analogous to that used in [3].

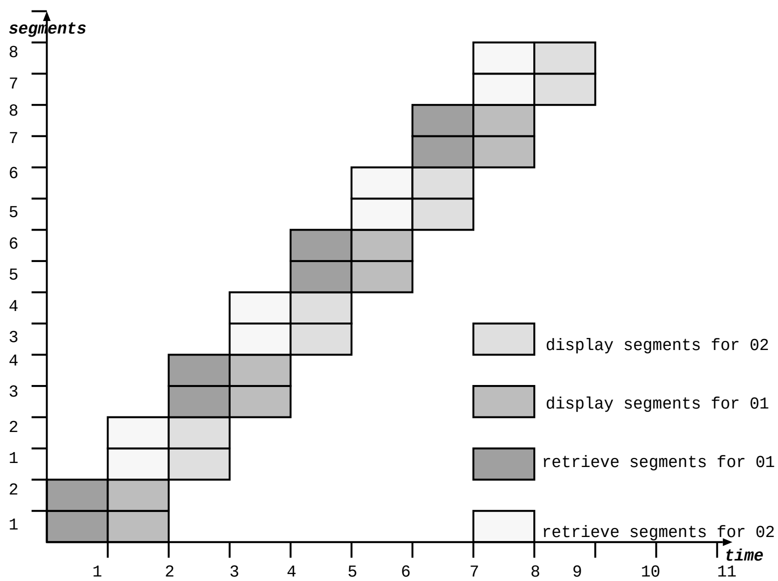

Figure 8. **Sample retrieval and display times of striped video segments**

Multiple Servers, Multiple Storage Devices (MSMSD)

An MSMSD VOD server consists of several server machines, each attached with several storage devices. For an MSMSD VOD server, several different configurations are possible. Two of these are configurations with:

1. a storage device dedicated to a single machine (e.g., Figure 5), and

2. a storage device accessible to every server machine in the configuration (e.g., Figure 9).

The first case illustrates a scaling of the SSMSD server system. The number of customers that can be supported increases by a multiple of the number of server

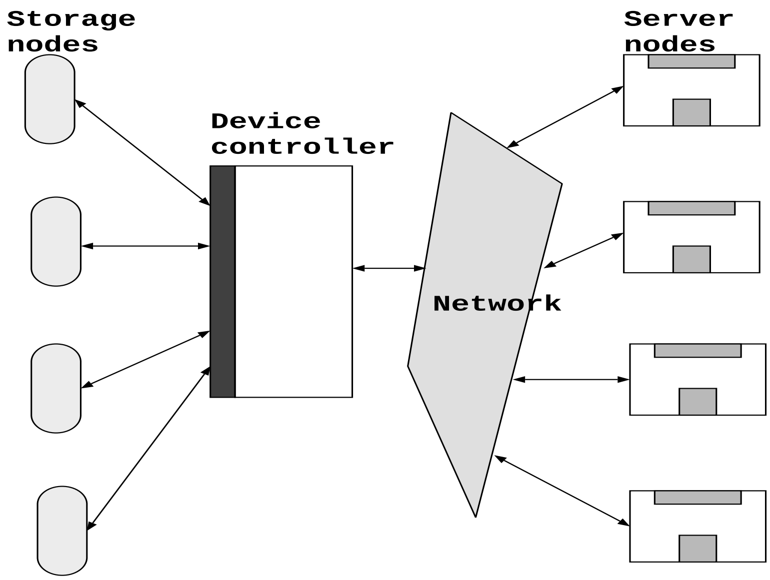

Figure 9. **An example of MSMSD system configuration**

nodes available. However, the response time for each customer request remains the same.

In the second case, the number of customers that can be concurrently supported is limited by the throughput of the interconnecting network; this assumes that the device controller can accommodate the cumulative transfer rate of the storage devices. The server nodes share the bandwidth of the network, therefore, if π Mb/s is the throughput of the network with ρ server nodes, then each node can service $\frac{\pi}{16\rho}$ customers concurrently (based on 16-segment video data). Therefore, the system can concurrently support $\frac{\pi}{\rho}$ customers. Since the video objects are striped, the response time is primarily determined by transmission time. Note that the scaling suggested in these configurations – SSMSD and MSMSD assume ideal situations. This is hardly the case, and in practice only a proportion of the theoretical scale up is actually achieved.

Another possible configuration is a situation in which the nodes connect directly to the access network and also to another network for inter-node communications. Each video request is accepted by a single server and the server alone handles the video stream. When a request comes through say $node_3$ for data stored in say $node_2$'s storage device, it must be routed to $node_2$ via the inter-node network. Similarly, $node_3$ may be servicing requests from other nodes. Consequently, the nodes are not only retrieving data, but also sending requests to other nodes and waiting for responses from other nodes. This level of functional complexity could adversely affect the response time of the system. When a video object is entirely stored in storage devices managed by a single server, the retrieval analysis is similar to that of the SSMSD. However, when objects are inter-node striped, then the response time becomes a more critical parameter because of the complexity of the system. This kind of overloaded responsibility increases the total I/O latency thereby increasing the data retrieval time and possibly reducing the throughput of the system.

4.2. DVDS clients

The DVDS server provides video storage and retrieval services for the DVDS clients. The clients include, but are not limited to, specialized processors, televisions, computers, etc. Some of the clients are capable of handling analog signals while others can handle digital data. There are different configurations for a DVDS and these generally differ from one business enterprise to another. A business enterprise may elect to directly attach end-user sites to the server(s) via a LAN as depicted in Figure 10. The client sites are most probably digital computers.

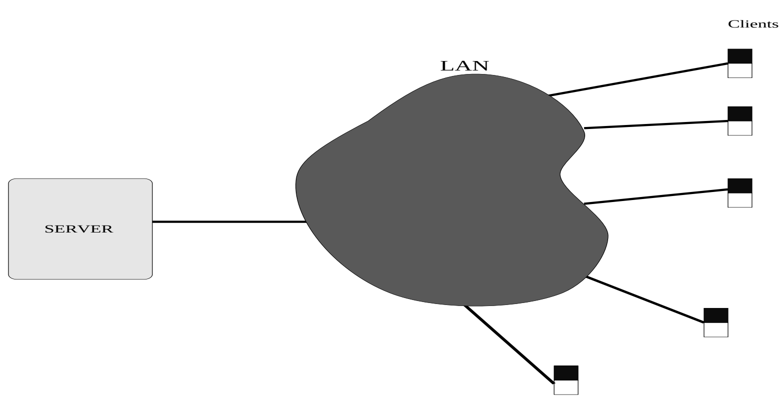

Figure 10. **Typical configuration for a business enterprise**

The types of clients utilized by companies that need to operate in more controlled environments like cable companies are different. These establishments control access to their equipment primarily for security. The physical location of the servers and a cable company control site (Hub) need not be the same. Figure 11 shows an example server architecture where the Hub is co-located with the server. The clients include televisions and digital equipment, e.g., computers.

When a server and the control site are not co-located, it will be necessary for them to be accessible to each other via some (public access) network, e.g., MAN, WAN. In such an architecture, a separate node may be used for access management and

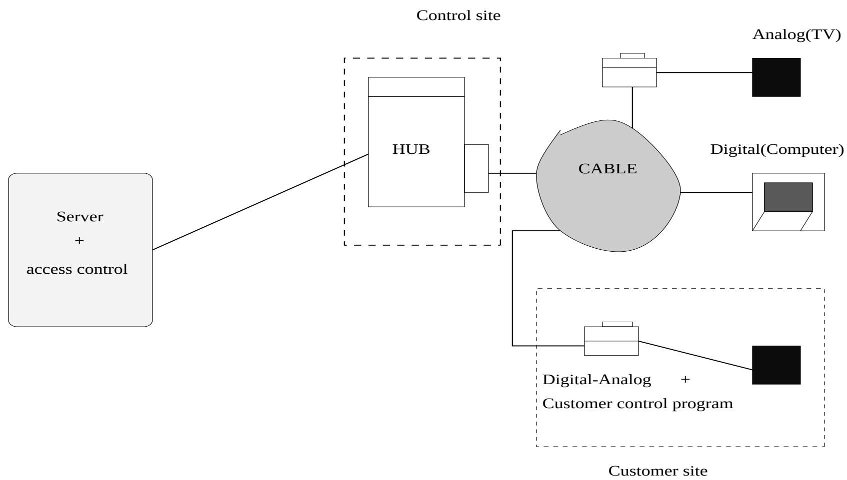

Figure 11. **An example configuration for a cable company. The server and the control site are co-located**

control, and different Hubs can be used to service different groups of subscribers, usually necessitated by geographical locations. In the presence of a network, we assume the existence of resource management capabilities in the Hubs. Requests coming from the users are authenticated by the access control node for such purposes as auditing and billing. Such a scenerio can be constructed from Figure 11, by attaching the Hub(s) to the public network, such that the server now connects to it (them) through the public access network, rather than directly.

4.3. DVDS connectivities

The reliability and efficiency of video data transport in a DVDS depends largely on the interconnectivities between the servers and clients. These are determined by the capabilities of the networks and device controllers. The differences in customer requirements and expectations and what the service providers can afford, play a significant role in determining the resource configurations of a DVDS environment.

Consider the connections between the server nodes and the storage devices. Recall from Figures 4, 5, and 6, that multiple storage devices can be attached to a single server to facilitate parallel access. To facilite this, the storage devices must be

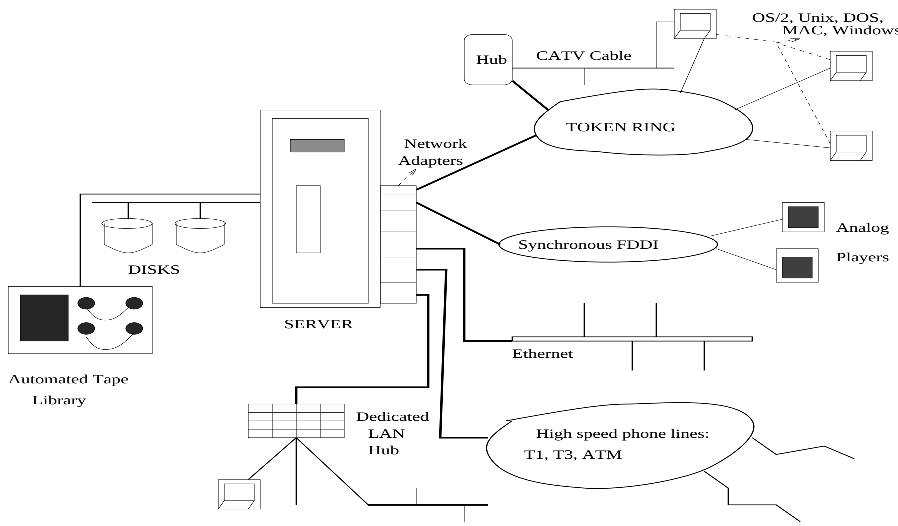

Figure 12. **An example environment configuration for a DVDS**

connected to the host machines via device controllers or adapters that support parallel transfers. We assume that these controllers and adapters can also sustain the maximum bandwidth required by these devices.

Reconsider Figure 5 where the server nodes are connected to a router node responsible for locating blocks of data. It is important that the router node is not allowed to become a bottleneck. The use of fast, high bandwidth networks such as FDDI network can be helpful in this respect. The server nodes in Figure 6 communicate through an interconnection network. The interconnection network is a non-blocking network that guarantees that every ONode can make a request to any INode. The network must be able to realize any non-conflicting request permutations between the ONodes and INodes.

An integrated environment for a DVDS (Figure 12) may consist of several nodes with both on-line and off-line storage devices attached to it. The off-line storage devices (tapes) are used to store infrequently accessed video programs.

4.4. Supporting CATV, PPV, VRC, and VOD services

We assume that Figure 12 is the general example of an environment for supporting the video services being discussed. As described in Section 2.1, CATV customers have the ability to view programs that are multicast to several customers at specific times. Each CATV customer can join an active program at any time during its

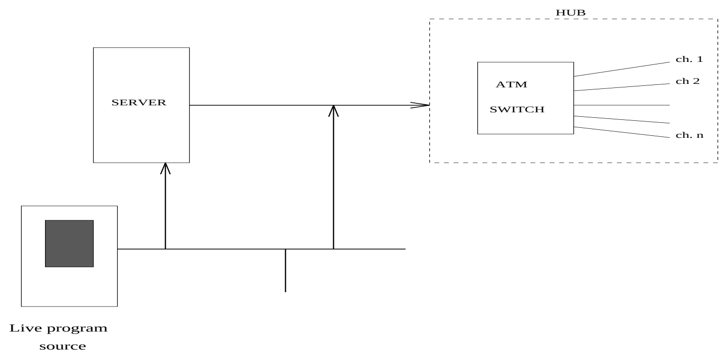

Figure 13. **Supporting CATV services in a DVDS**

broadcast session. In order to provide DVDS customers the same functions, it is necessary to employ a multicast device that can broadcast a given program. Figure 13 shows an ATM (Asynchronous Transfer Mode) Switch in a Hub which replicates a given ATM cell for the multicast channels. A video program data flows from the server to the switch where it is replicated and transmitted through different channels from which the customers can view the program. Control utilities are employed to control the transmission of a given program with respect to its start and end times.

Since these types of activities are pre-scheduled, they are buffered at appropriate sites to minimize response times. The expected delay is equal to the time of changing from one channel to another. It is also possible to obtain different image qualities since ATMs are flexible enough to allow the transport of both constant and variable rate services. The DVDS must also be able to provide real-time (live) programs to its customers. In that case, it is necessary to transmit the live program to both the control and server sites (see Figure 13). The server stores the program for later viewing by the customers. The provision of PPV service by the DVDS is not different from that of the CATV. The same configuration and protocols apply. The determination as to when a program is broadcast depends on its schedule. The same setup is also useful for satisfying VRC service. However, in this case, it may be necessary for the customer to wait for a relatively long time for an archived program to become available. Multicasting is not usually applicable in this scenario since it is not very likely for different customers to request the same video at exactly the same time.

The configuration depicted in Figure 13 can also support VOD services. Customers can view live or recorded programs on request. Using the application programs on the control devices, they can stop, forward, backward, or pause during their viewing session. Multicasting or broadcasting becomes unattractive in VOD services because of the obvious temporal differences in customer requests.

Table 3. **DCL (top half) and SDCL (bottom half) placement schemes**

Object Segments					
d0	d1	d2	d3	d4	d5
X0.0	X0.1	X0.2	X0.3	X0.4	X0.5
X0.6	X0.7	X0.8	X0.9	X0.10	X0.11
X0.12	X0.13	X0.14	X0.15	X0.16	X0.17
X0.18	X0.19	X0.20	X0.21	X0.22	X0.23
X0.24	X0.25	X0.26	X0.27	X0.28	X0.29
X0.0	X0.1	X0.2	X0.3	X0.4	X0.5
X0.11	**X0.6**	X0.7	X0.8	X0.9	X0.10
X0.16	X0.17	**X0.12**	X0.13	X0.14	X0.15
X0.21	X0.22	X0.23	**X0.18**	X0.19	X0.20
X0.26	X0.27	X0.28	X0.29	**X0.24**	X0.25

5. Data Layout in Interactive Video-on-Demand Systems

VOD applications require interactivity in the form of stream playout control that allows a user to do *fast forward (ff), rewind (rw), slow play, slow rewind, pause*, and *stop-and-return* on a media stream as is done in video cassettes. A user may also access the media streams in a random manner. These operations have implications for data layout and scheduling in an interactive video-on-demand (IVOD) environment. In the next section, we discuss two data layout algorithms, the Distributed Cyclic Layout and the Staggered Distributed Cyclic Layout and use these in a mirroring scheme to provide improved performance in a video-on-demand system.

5.1. Data layout algorithms

A number of multimedia data layout schemes were described in [4]. Two of these are the *Distributed Cyclic Layout (DCL)* and the *Staggered Distributed Cyclic Layout (SDCL)*. Table 3 shows an example. These are essentially forms of disk striping [10], and while they increase the I/O throughput of multimedia systems, they provide no fault tolerance. We note that DCL is basically analogous to the *modulo D* algorithm (D is number of disks in array), while SDCL is a modified version of DCL where data placement for the next stripe starts at disk $(k + s)\, modulo\, D$ given that the current stripe starts at disk k. s is the stagger distance. In Table 3, $s = 1$. SDCL is also similar to *staggered striping* introduced in [3] and chained declustering [15].

There are some problems with the DCL scheme. Consider the video frames stored in $D = 6$ nodes using the DCL algorithm as shown in Table 3. In normal playout, the frame sequence is X0.0, X0.1, X0.2, X0.3, X0.5, X0.6, ..., whereas for *ff* the frame sequence could be X0.0, X0.2, X0.4, X0.6, X0.8, X0.10, ... Since data layout is with DCL algorithm, the set of nodes visited during normal playout is 0,1,2,3,4,5, ..., whereas in *ff* mode the nodes visited are 0,2,4,0,2,4, ... The problem with this is that the stream control alters the sequence of node-visits from the normal linear (modulo D) sequence. Moreover, it creates "hot-spots" and in turn requires bandwidth to be reserved at each node to deal with the overloads.

Note that if the SDCL scheme suggested in Table 3 is used, there would not be the hot-spots problems observed with the DCL scheme. For example, reconsider the *ff* sequence X0.0, X0.2, X0.4, X0.6, X0.8, X0.10, ... From Table 3 we observe that with SDCL the nodes visited are 0,2,4,1,3,5.

5.2. Declustered mirror

Table 4. **Cluster 0 of declustered mirror. Cluster size = 6. DCL algorithm is used for this primary copy cluster.**

d0	d1	d2	d3	d4	d5
X.0	X.1	X.2	X.3	X.4	X.5
X.6	X.7	X.8	X.9	X.10	X.11
X.12	X.13	X.14	X.15	X.16	X.17
X.18	X.19	X.20	X.21	X.22	X.23
X.24	X.25	X.26	X.27	X.28	X.29
X.30	X.31	X.32	X.33	X.34	X.35
X.36	X. 37	X.38	X.39	X.40	X.41
X.42	X.43	X.44	X.45	X.46	X.47
X.48	X.49	X.50	X.51	X.52	X.53
X.54	X.55	X.56	X.57	X.58	X.59

Table 5. **Cluster 1 of declustered mirror. Cluster size = 6. SDCL algorithm is used for this backup copy cluster.**

d6	d7	d8	d9	d10	d11
X.0	X.1	X.2	X.3	X.4	X.5
X.11	**X.6**	X.7	X.8	X.9	X.10
X.16	X.17	**X.12**	X.13	X.14	X.15
X.21	X.22	X.23	**X.18**	X.19	X.20
X.26	X.27	X.28	X.29	**X.24**	X.25
X.31	X.32	X.33	X.34	X.35	**X.30**
X.36	X.37	X.38	X.39	X.40	X.41
X.47	**X.42**	X.43	X.44	X.45	X.46
X.52	X.53	**X.48**	X.49	X.50	X.51
X.57	X.58	X.59	**X.54**	X.55	X.56

Declustered mirror is a novel architecture that allows for the provision of a wide range of fast forward and rewind speeds under a load balance condition. Declustered mirror is derived from:

- *Disk stripping* [5, 10], to provide high I/O bandwidth for multimedia data.
- *Distributed Cyclic Layout (DCL)* and *Staggered Distributed Cyclic Layout (SDCL)* [4] to provide a data layout scheme which supports a wide range of fast forward

and rewind speeds under load balance condition for interactive video-on-demand systems.

- *Distorted mirroring* [22] to provide fault tolerance against a single disk failure.
- *Chained Declustering* [15] to provide load balance during failure mode operation.

Tables 4 and 5 show an example of data layout in a declustered mirror scheme. There are $N = 12$ disks, numbered $0, 1, \ldots, 11$. These disks are divided into two clusters as suggested in chained declustering [15]. Cluster 0 consists of disks $0, 1, \ldots, 5$ while cluster 1 consists of disks $6, 7, \ldots, 11$. The clusters form a mirror pair in a manner almost analogous to chained declustering. All disks in a cluster form one logical disk, with the logical disk of cluster 0 holding the primary copy of data and the backup copy held in the logical disk that constitutes cluster 1. To simplify exposition, we assume one large object **X** with sixty fragments $X.0, X.1, \ldots, X.59$ placed in the disks as shown in Tables 4 and 5. Data in the primary logical disk is allocated using the DCL algorithm, while SDCL is used in the backup logical disk. Note that mirroring protection refers to a logical disk. For example logical disk 0 is mirrored on logical disk 1. In the next section, we present preliminary results of an on-going study exploiting the advantages of these schemes to improve performance in an IVOD environment.

6. Simulation Experiment

In a practical declustered mirror application, only the hot set of data is mirrored. For example, consider a VOD housing 1500 movies. Due to cost considerations, all the movies would be resident on a tertiary storage, while only a small percentage are cached on disks. For purposes of illustration, we assume that only 10% of these are on disk. These 150 movies would include a small proportion of frequently demanded movies – the hot set. We assume that there are about 20 of these. Figure 14 illustrates the placement of the movies in the two sets of disks. 500 disks in two groups of 250 disks are used. The first group consists of disks number 0 to 249 while the second group consists of the remaining disks – number 250 to 499. All the 1500 movies are resident on tape; 20 of the 150 movies on disk constitute the hot set and are mirrored; the remaining 130 are not mirrored.

In [23] we observed that the DCL algorithm provides load-balance for speeds that are prime to the number of disks (nodes) D, while SDCL provides load-balance for speeds that are factors of D. What is observed is that as D increases, the numbers that are prime to D increase faster than the numbers that are factors of D. In fact, the factors of D fluctuate a lot and in the particular cases when D is prime, no factors are obtained with the result that the SDCL scheme does not contribute to the number of variable speeds supported except at speed $= D$, the cluster size. This observation leads us to conclude that for installations with many disks, it is probably better to use DCL as the dominant storage scheme. However, in Figure 14 we used SDCL to mirror the hot set and DCL for the other movies. This decision

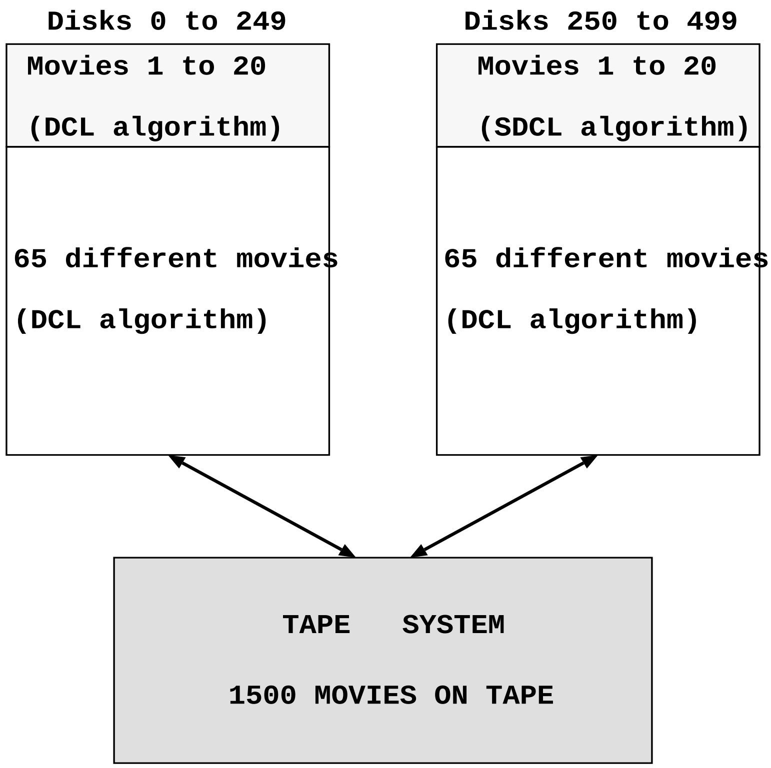

Figure 14. **Mirroring of hot set using declustered mirroring**

to use SDCL for mirroring was ad-hoc. In our continuing research we will formally determine whether the hot set of movies should be mirrored using the SDCL or DCL scheme. As already noted, we assume a total of 1,500 movies on the tertiary storage; 10% of these – 150 movies are on disk, and about 13% of these 150 movies – 20 movies, constitute the hot set and are mirrored. To simplify the exposition we assume that movies 1 to 20 constitute the hot set.

We are currently evaluating the costs and benefits of declustered mirroring using simulation. Table 6 summarizes the simulation parameters of one set of experiments. All movies were assumed to be of equal duration – 100 minutes. In this experiment, the replacement policy was not an issue since the movies that constitute the hot set were always on disk. For those few instances when the desired movies were not on disk, the system ejected the least recently used movie from the disk.

The results presented here are for two schemes:

1. *One group:* In this scheme, all the 500 disks were used to store unique movies.

Table 6. **Summary of simulation parameters**

Tape drive	1
Disks	500
Disk capacity	1 Gbytes
Size of each movie	3 Gbytes
Probability of interactivity	Between 10% & 90%
Number of movies on tape	1500
Number of movies on disk	150
Number of hot set movies	20

2. *Declustered mirror - DCL/SDCL:* This is the allocation illustrated in Figure 14.

In Figures 15 and 16 we show the throughput observed for the two schemes studied. In both figures, the horizontal axis shows the probability (in percentage) that a user performs a fast forward or rewind operation. In the simulation, a user admitted into the system resides there for a randomly generated period of time. For this study the time varies between 30 minutes and 4 hours. During this residence time, with a certain probability, the user requests frames from the server in a manner that represents normal playback, and fast forward or rewind operation at randomly determined speeds. The user exits the system at the expiration of its residence time.

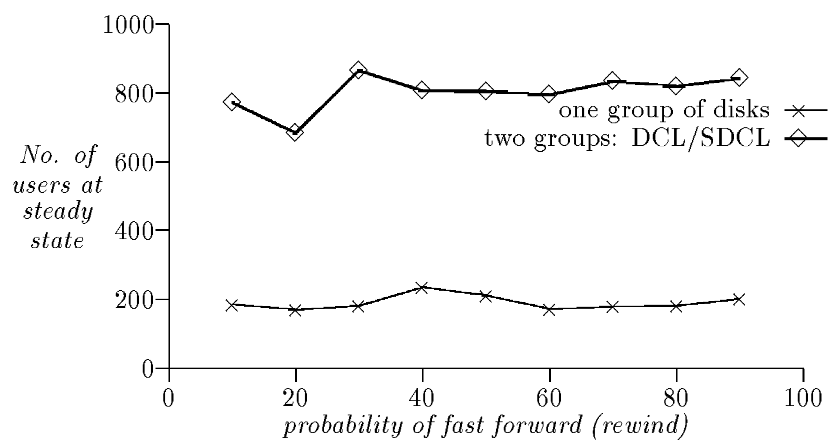

Figure 15. **Steady state throughput vs. probability of interactivity**

Figure 15 is the steady state number of users in the system, while Figure 16 is the maximum number of users observed in the system during the simulation. As both curves illustrate, declustered mirror (DCL/SDCL) clearly outperforms the single disk configuration.

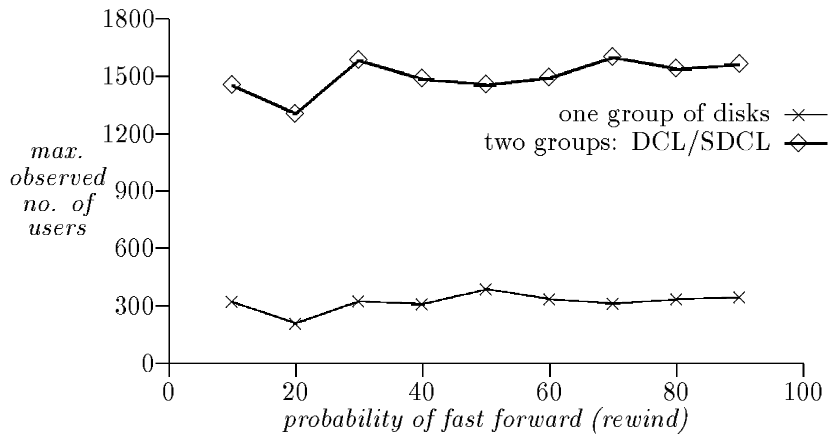

Figure 16. **Maximum throughput achieved vs. probability of interactivity**

The reason for the better performance can be explained as follows. The system performance is limited by available bandwidth of the disks. Since about 90% of the accesses are to the hot-set of movies which are almost always on disk, tape access is rare. The result is that the system with mirrored hot set potentially has double bandwidth and hence can service more users. Thus a better performance is observed relative to the system with one set of disks. It is, however, important to note that we have assumed that the hot set is on-disk in both cases. This conclusion may change if tape accesses increase indicating the absence of the hot set movies on disk. In such a situation, the single disk system is more likely to have a greater proportion of the hot set movies on-disk reducing the number of accesses to tape. However, for the scenario shown in Figures 15 and 16, the performance of the DCL/SDCL scheme was even much better than expected. This is because not only could more users be serviced, but the SDCL scheme allowed those fast forward and rewind speeds that DCL could not service without violating load balance conditions.

7. Conclusion

In an effort to provide an integrated design consideration for a Digital Video Delivery System (DVDS), we presented the architectural descriptions and the types of conventional services available to video customers through non-digital means. These systems/services include the Cable Television, Pay-Per-View, and Video Repository Centers (rental). We also discussed the requirements for Video On Demand services. The possible configurations of a DVDS were presented using three general system components - servers, clients, and connectivities. An integrated architecture was proposed for a DVDS that addresses the requirements of a conventional video customer. We also presented preliminary results of an ongoing study to exploit the characteristics of *Distributed Cyclic Layout (DCL)* and *Staggered Distributed Cyclic Layout (SDCL)* data allocation schemes to improve performance of an IVOD system. The discussions in this paper are not based on any specific hardware or software products, rather, the general technical requirements of a DVDS were presented and analyzed.

References

1. R. Aravind, G. Cash, Duttweiler, and H. Hang, "Image and Video Coding Standard", AT&T Technical Report, 1993.
2. S. Battista, S. Pereira, C. Pires, and G. Salema, "MPEG Picture Coding Standards: An Evolution Towards Computer Image Applications", *Proceedings, 2nd International Conference on Computational Graphics and Visualizations Techniques*, December 1992, Libson, Portugal, pp. 230-236.
3. S. Berson and S. Ghandeharizadeh, "Staggered Striping in Multimedia Information Systems", *Proceding, International Conference of the ACM SIGMOD*, Minneapolis, Minnesota, May, 1994.
4. M. Buddhikot and G. Paulkar, "Distributed data Layout, Scheduling and Playout Control in a Large Scale Multimedia Storage Server", Technical Report WUCS-94-33, Department of Computer Science, Washington University, St Louis, MO 63130,1994.
5. P. Chen, G. Gibson, R. Katz and D. Patterson, "A Case for Redundant Arrays of Inexpensive Disks (RAID)", *Proceedings, International Conference of the ACM SIGMOD*, June 1988, Chicago, Illinois, pp. 109-116.
6. E. Fox, "Advances in Interactive Digital Multimedia Systems", *IEEE Computer*, pp. 9-21, 1991.
7. B. Furht, D. Kaira, F. Kitson, A. Rodriguez, and W. Wall, "Design Issues for Interactive Television Systems", *IEEE Computer*, pp. 25-38, 1995.
8. A. Gelman, H. Kobrinski, L. Smoot, S. Weinstien, M. Fortier, and D. Lemay, " A Store and Forward Architecture For Video-on-Demand Service", *Procedings, International Conference on Communications*, Volume 2, pp. 842-846, Denver,CO, June 1991.
9. D. Gemmell, H. Vin, D. Kaundlur, P. Rangan, and L. Rowe, "Multimedia Storage Servers: A Tutorial", *IEEE Computer* pp. 40-49, May 1995.
10. H. Garcia-Molina, and K. Salem, "Disk Striping", *Proceedings, IEEE International Conference on Data Engeering*, February 1986, Los Angeles, CA., pp. 336- 345.
11. H. Gharavi, "Subbanding Coding of Video Signals", In J. Woods, editor, *Subband Image Coding*, Kluwer Academic Publishers Group, 1990.
12. H. Gharavi, "Subbanding Coding Algorithm for Video Applications: Videophone to HDTV Conferencing", *IEEE Transctions on CAS for Video Technology*, pp. 174-183, June 1991.
13. W. Hogde, S. Mabon, and J. Powers, "Video-on-Demand: Architecture, Systems, and Applications", *SMPTE Journal*, 102(9):791-803, Semtember 1993.
14. C. Horne, and A. Puri, "Video Cooling With Adaptive Quantization and Rate Control", *Proceedings, SPIE*, volume 1818, 1992, pp. 798-806.
15. H. Hsiao, and D. DeWitt, "Chained Declustering: A New Availability Strategy for Multiprocessor Database Machines", *Proceedings, IEEE International Conference on Data Engineering*, February 1990, Los Angeles, California, pp. 456-465.
16. H. Kuroda, M. Fujimura, Y. Natoachi, and M. Ikehara, "Variable Bit Rate Video Coding Method", *Electronics Communication*, Apri 1991, pp. 46-58.
17. D. LeGall, "MPEG: A Video Compression Standard for Multimedia Applications", *Communications of the ACM*, April 1991, pp. 46-58.
18. M. Liou, "Overview of the PX64 Kbit/s Video Coding Standard", *Communications of the ACM*, April 1991, pp. 59-63.

19. M. Maruyama, H. Sakamoto, Y. Ishibashi, and K. Nishimura, "High-speed Hardware Architecture for High-definition Videotex System", *Proceedings, SPIE*, 1991, pp. 916-927, Volume 1605.
20. F. McDevitt, and J. Sutherland, "Switched vs Broadcast Video for Fiber to the Home Systems", *Proceedings, International Conference on Communications*, April 1990, Atlanta, Ga., pp. 1109-1119.
21. K. Nwosu, "Data Storage Modeling and Management for Multimedia Information Systems, PhD thesis, December 1993 Syracuse University, Syracuse New York.
22. C. Orji, and J. Solworth, "Distorted Mirrors", *Proceedings, First International Conference on Parallel and Distrubuted Information Systems*, December 1991, Miami, FL., pp. 10-17.
23. C. Orji, K. Nwosu, N. Rishe, "Data Layout for Interactive Video-on-demand Storage Systems", *Proceedings, Eighth International Conference on Software Engineering and Knowledge Engineering*, Lake Tahoe, Nevada, 1996.
24. S. Ramanathan, and P. Rangan, "System Architectures for Personalized Multimedia Services", *IEEE MultiMedia*, Spring 1994, 1(1):37-46.
25. V. Ramamoorthy, and R. Ramarao, "Architectural Design of on Demand Video Delivery Systems: The Spatio Temporal Storage Allocation Problem", *Proceedings, International Conference on Communications*, June 1991, Denver, CO., pp. 506-510.
26. W. Twabi, F. Horn, E. Horlatt and J. Stefani, "Video Compressions Standards and Quality of Service", *The Computer Journal*, 36(1):43-54, 1993.
27. G. Wallace, "The JPEG Still Picture Compression Standard", *Communications of the ACM*, pp. 30-44, April 1991.